TCP/IP communication between two USRP-E110

Luis Sanabria-Russo
Technical Report
luis.sanabria@upf.edu

Abstract—**This short report intends to provide an overview of the procedure and statistics of establishing a TCP/IP link between two USRP-E110. The testings are performed using an example GNURadio code and the networking protocol stack provided by the Linux operating system embedded in the USRP-E110.**

I. INTRODUCTION

During the past four months the group has been dedicating efforts towards the understanding and how to implement some of the features included in GNURadio [1]. Our testings resulted in an effective observation of the TV UHF spectrum that allowed the implementation of a TV White Spaces (TVWS) detector using energy detection technique [2], [3].

Our current challenge relates to transmission over these detected TVWS. Fortunately, GNURadio provides a set of example code that can me used for testings and modified to achieve desired tasks.

II. GNURADIO CODE

The example code used to achieve network-like communication through the radio link established by the USRP-E110s is called `tunnel.py`. It is usually located at */gnuradio-examples/digital/narrowband* of the GNURadio root directory.

`tunnel.py` builds a TAP interface [4] in the kernel for tunneling Ethernet frames and uses the Operating System (OS) network protocol stack.

This TAP is associated with a virtual interface that is used to test IP connectivity. Figure 1 provides an overview of the setup used for the testings.
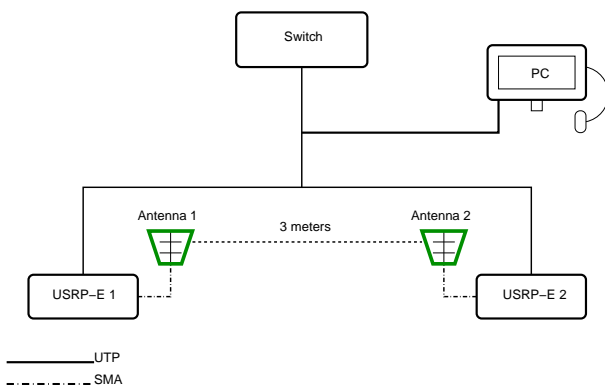

Fig. 1. Testings setup.

The execution of the code requires access to both USRP-E110s (either via USB console port or SSH). Once inside the following parameters are passed to `tunnel.py`:

```
./tunnel.py -f 560e6 -r 0.1e6
--rx-gain=20 --tx-gain=20 -m gmsk -p 2
--omega-relative-limit=0.00 5
--gain-mu=0.175 --mu=0.5
```

, where 560 MHz, 100 kbps, 20 dB, 20 dB represent the center frequency, rate, receiver and transmitter gains accordingly. The rest of the parameters are related to PHY characteristics [5].

Then, at each USRP-E110 a virtual interface (usually `gr0`) should be configured with an IP address inside the same subnet by issuing the command:

```
sudo ifconfig gr0 <ip address>
```

Conventional IP connectivity should be established between the two devices.

III. RESULTS AND FUTURE WORK

With the current approach results show around 13% packet loss, delays around 60 ms at a throughput of 100 kbps. Also, it is possible to perform higher-layer communication with `ssh` logins and remote management of the devices.

Being able to replicate this exercise allows for experimentation with medium access technologies and radio links. Future testings are pending in order to gather more accurate statistics and reduce packet loss.

REFERENCES

[1] Blossom, E., "GNU radio: tools for exploring the radio frequency spectrum," *Linux J.*, vol. 2004, no. 122, pp. 4–, Jun. 2004. [Online]. Available: http://dl.acm.org/citation.cfm?id=993247.993251

[2] Sanabria-Russo, L., Barcelo, J., Bellalta, B., Domingo, A., "Spectrum Sensing with USRP-E110," July 2012, submitted.

[3] Cabric, D. and Tkachenko, A. and Brodersen, R.W., "Experimental study of spectrum sensing based on energy detection and network cooperation," in *Proceedings of the first international workshop on Technology and policy for accessing spectrum.* ACM, 2006, p. 12.

[4] Krasnyansky, M., Yevmenkin, M., "Virtual Point-to-Point (TUN) and Ethernet (TAP) devices." [Online]. Available: http://vtun.sourceforge.net/tun/faq.html

[5] GNURadio Project, "Bug report: benchmark_tx.py / benchmark_rx.py not working on the air." [Online]. Available: http://gnuradio.org/redmine/issues/196